Dokumentation

Documentation for robotlibrary/robot.py

Robot

This is the central class which manages and uses all the other components of the robot. The parameters are defined in config.py

_drive

This abstracted driving function is only called locally by the other functions with better names. It accelerates and decelerates to make driving more natural. Do not call directly!

_drive_instantly

This abstracted driving function is only called locally by the other functions with better names. It sets the speed immediately. Do not call directly!

set_speed_instantly

Sets the new speed immediately. Doesn't change the driving mode of the robot. :param s: the speed you want to set.

set_speed

Sets the new speed and accelerates and decelerates. Doesn't change the driving mode of the robot. :param s: the speed you want to set.

set_forward

Sets the direction of the robot. True means forward. :param f: True for forwards and False for backwards.

spin_right

Spin right indefinitely.

spin_left

Spin left indefinitely.

turn_right

This turns the robot to the right without it spinning on the spot. Each call makes the turn steeper.

turn_left

This turns the robot to the right without it spinning on the spot. Each call makes the turn steeper.

go_left

With Meccanum wheels the robot goes sideways to the left.

go_right

With Meccanum wheels the robot goes sideways to the right.

turn

This turns the robot right or left. Is mostly used by the remote control. :param turn: positive or negative value. Higher values mean steeper turn.

go_straight

Lets the robot go straight on. Usually called when a turn shall end.

spin_before_obstacle

This spins until the distance to an obstacle is greater than the given parameter *distance*. :param distance: The distance

toggle_spin

Toggle turn for the given duration. With each call the opposite direction(clockwise / anti-clockwise) is used. :param d: The duration for the turn in milliseconds.

random_spin

Randomly turn for the given duration. :param d: The duration for the turn in milliseconds.

stop

Stop the robot slowly by deceleration.

emergency_stop

Stop the robot immediately.

ir_detected

If implemented this method is called when the IR-sensor has detected a change. Fill in your code accordingly.

get_dist

Get the distance from the ultrasonic sensor.

set_angle

If implemented, turn the servo motor with the ultrasonic sensor to the given angle. :param a: The angle that is to be set.

get_smallest_distance

This returns the angle of the ultrasonic sensor where it measured the smallest distance

get_longest_distance

This returns the angle of the ultrasonic sensor where it measured the longest distance

Documentation for robotlibrary/config.py

Module

This defines the parameters for the motors.

MAX_DUTY: Set to lower than the maximum not to overload the motors. Absolute maximum is 65535. MIN_DUTY: Set this to the minimum duty cycle that the motor needs to start moving. MIN_SPEED: Only 0 is making sense here but if you want you can change that. Must be above 0 though. MAX_SPEED: If you want another scale than 0-100, set the maximum here.

DEBOUNCE_WAIT: This defines the waiting time for the debouncing of the buttons. Leave as it is if you don't know what it means.

WHITE_DETECTED: Use these constants to check for white or black with the IR-sensor. Don't change! BLACK_DETECTED: Use these constants to check for white or black with the IR-sensor. Don't change!

Motors and ultrasonic sensor must use consecutive pins. so, f. ex. the left motor uses pins 12 and 13. Use >None< if you don't use the device. MLF and LRF are for four wheel drive. ML: pin number

for left motor (or left rear motor for four wheel drive). MR: pin number for right motor (or right rear motor for four wheel drive). MLF: pin number for left motor (or left front motor for four wheel drive). Use None if not used. MRF: pin number for right motor (or right front motor for four wheel drive). Use None if not used. US: pin number for the ultrasonic sensor. Use None if not used. IR: pin number for the infrared sensor. Use None if not used. SERVO: pin number for the servo motor. Use None if not used.

JS_X_MEDIAN JS_Y_MEDIAN JS_MAX_DUTY JS_MIN_DUTY: These define the parameters for the joystick. You need to calibrate the numbers. Look at joystick.py for details.

ROBOT_NAME: You need to set a custom name if you use a remote control.

SERVO_MIN_DUTY: Only change if the servo doesn't move the required 180°. SERVO_MAX_DUTY: Only change if the servo doesn't move the required 180°.

Revision #4
Created 22 May 2025 11:36:24 by Marcus Jacobs
Updated 22 May 2025 12:51:10 by Marcus Jacobs